CLAIM AMENDMENTS

1. (Previously Presented) A method for encryption and decryption of electronic messages based on an encryption protocol, the method comprising the computer-implemented steps of:

receiving a first electronic message that is encrypted according to the encryption protocol;

generating at least one part of a second electronic message, based on at least the first electronic message, a modular operation that is based on two applications of Montgomery's method, a first operand, a second operand, a modulus, and a modular reduction using a negative multiplicative inverse of the modulus, and wherein the step of generating the second electronic message includes the computer-implemented steps of:

generating a first constant (R) based on the modulus (M) by selecting a second constant (W) such that $W \geq 4M$ and determining the first constant (R) according to the expression $R = W^2$ (mod(M));

in a first application of Montgomery's method, determining an intermediate result based on at least Montgomery's method for the modular operation, the first operand, and the first constant; and

in a second application of Montgomery's method, determining and storing in memory a final result that comprises the at least one part of the second electronic message, based on at least Montgomery's method for the modular operation, the intermediate result, and the second operand.

2-6. (Cancelled)

7. (Previously Presented) The method of Claim 1, wherein the second constant (W) is not a power of two.

8. (Cancelled)

9. (Currently Amended) The method of Claim 1, wherein the modular operation for determining the intermediate result is a modular multiplication that is based on at least the first operand, the first constant, a second constant, the modulus, and ~~a~~ the negative multiplicative inverse of the modulus, and wherein the intermediate result is determined by the computer-implemented steps of:

determining a modified first operand based on the first operand and the first constant;

determining a modular reduction of the modified first operand based on the modified first operand, the negative multiplicative inverse of the modulus, and the second constant; and

determining the intermediate result based on the modified first operand, the modular reduction of the modified first operand, the modulus, and the second constant.

10. (Cancelled)

11. (Previously Presented) The method of Claim 1, wherein the modular operation for determining the final result is a modular multiplication that is based on at least the second operand, a second constant, the modulus, the negative multiplicative inverse of the modulus, and the intermediate result, and wherein the final result is determined by the computer-implemented steps of:

determining a modified second operand based on the second operand and the intermediate result;

determining a modular reduction of the modified second operand based on the modified second operand, the negative multiplicative inverse of the modulus, and the second constant; and

determining the final result based on the modified second operand, the modular reduction of the modified second operand, the modulus, and the second constant.

12. (Cancelled)

13. (Previously Presented) The method of Claim 1, wherein the modular operation for determining the intermediate result is a modular exponentiation that is based on at least the first operand, the first constant, a second constant, the modulus, and the negative multiplicative inverse of the modulus, and wherein the intermediate result is determined by the computer-implemented steps of:

determining a modified first operand based on the first operand and the intermediate result;

determining a modular reduction of the modified second operand based on the modified second operand, the negative multiplicative inverse of the modulus, and the second constant; and

determining the intermediate result based on the modified second operand, the modular reduction of the modified second operand, the modulus, and the second constant.

14. (Cancelled)

15. (Previously Presented) The method of Claim 1, wherein the modular operation for determining the final result is a modular exponentiation that is based on at least the second operand, a second constant, the modulus, the negative multiplicative inverse of the modulus, and the intermediate result, wherein the second operand includes a plurality of digits, and wherein the final result is determined by the computer-implemented steps of:

specifying a previous final result as having a value of one and a previous intermediate result as the intermediate result;

for each digit of the plurality of digits included in the second operand, performing the computer-implemented steps of:

when each digit of the plurality of digits has the value of one, then performing the computer-implemented steps of:

determining an updated final result based on the previous final result and the previous intermediate result;

determining a modular reduction of the updated final result based on the updated final result, the negative multiplicative inverse of the modulus, and the second constant;

determining a revised updated final result based on the updated final result, the modular reduction of the final result, the modulus, and the second constant;

if all digits of the plurality of digits have not been evaluated, specifying the revised updated final result as the previous final result; and

if all digits of the plurality of digits have been evaluated, specifying the revised updated final result as the final result;

determining an updated intermediate result based on the previous intermediate result;

determining a modular reduction of the updated intermediate result based on the updated intermediate result, the negative multiplicative inverse of the modulus, and the second constant;

determining a revised updated intermediate result based on the updated intermediate result, the modular reduction of the updated intermediate result, the modulus, and the first constant; and

specifying the revised updated intermediate result as the previous intermediate result.

16. (Cancelled)

17. (Original) The method of Claim 1, further comprising the computer-implemented steps of:

generating a plurality of residual number system (RNS) representations, wherein the plurality of RNS representations includes at least one RNS representation for each of the first operand, the modulus, and the first constant;

wherein the step of determining the intermediate result includes the computer-implemented step generating the intermediate result based on Montgomery's method for the modular operation and the plurality of RNS representations; and

wherein the step of determining the final result includes the computer-implemented step of generating the final result based on Montgomery's method for the modular operation and the plurality of RNS representations.

18. (Previously Presented) The method of Claim 17, wherein:

the plurality of RNS representations includes a first set of RNS representations in a first RNS base and a second set of RNS representations in a second RNS base;

the first RNS base extends the second RNS base;

the first RNS base includes a first group of sixty-four residues and the second RNS base includes a second group of sixty-four residues;

each residue in both the first group and second group is a seventeen-bit residue;

each residue in the first group is relatively prime with respect to all other residues in the first group and each residue in the second group is relatively prime with respect to all other residues in the second group;

each residue in both the first group and the second group are a selected from a range of 2^{16} to 2^{17}; and

the method further comprises the computer-implemented steps of:

converting, in eight clock cycles, a first RNS representation of the first set of RNS representations in the first RNS base to a second RNS representation of the second set of RNS representations in the second RNS base;

performing operations involving each residue in the first group in parallel; and

performing operations involving each residue in the second group in parallel.

19-26. (Cancelled)

27. (Previously Presented) The method of Claim 1, wherein the modular operation is a modular multiplication and the step of generating the second electronic message further includes the computer-implemented step of:

while determining the intermediate result and determining the final result, storing results of intermediate computations in a first register file and a second register file, wherein the first register file includes a first set of sixty-four seventeen-bit registers and the second register file includes a second set of sixty-four seventeen-bit registers.

28. (Cancelled)

29. (Previously Presented) The method of Claim 1, wherein the modular operation is a modular exponentiation and the step of generating the second electronic message further includes the computer-implemented step of:

while determining the intermediate result and determining the final result, storing results of intermediate computations, in a first register file a second register file, a third register file, and a fourth register file; and

wherein the first register file includes a first set of sixty-four seventeen-bit registers, the second register file includes a second set of sixty-four seventeen-bit registers, the third register file includes a third set of sixty-four seventeen-bit registers, and the fourth register file includes a fourth set of sixty-four seventeen-bit registers.

30. (Cancelled)

31. (Previously Presented) The method of Claim 1, wherein the steps of determining the intermediate result and determining the final result use an array of sixty-four seventeen-bit by seventeen-bit modular multiplier circuits, and wherein the array of sixty-four seventeen-bit by seventeen-bit modular multiplier circuits includes a plurality of ratio four to two compressors that are organized into three levels and that are executed in one clock cycle.

32. (Cancelled)

33. (Original) The method of Claim 1, wherein the steps of determining the intermediate result and determining the final result use an array of sixty-four thirty-four-bit to seventeen-bit modular reduction circuits that are executed in one clock cycle.

34. (Previously Presented) A computer-readable medium carrying one or more sequences of instructions for encryption and decryption of electronic messages based on an encryption protocol, which instructions, when executed by one or more processors, cause the one or more processors to carry out the steps of:

receiving a first electronic message that is encrypted according to the encryption protocol;

generating at least one part of a second electronic message, based on at least the first electronic message, a modular operation that is based on two applications of Montgomery's method, a first operand, a second operand, a modulus, and a modular reduction using a negative multiplicative inverse of the modulus, and wherein the instructions for generating the second electronic message further comprise instructions which, when executed by one or more processors, cause the one or more processors to carry out the steps of:

generating a first constant (R) based on the modulus (M) by selecting a second constant (W) such that $W \geq 4M$ and determining the first constant (R) according to the expression $R = W^2 \pmod{M}$;

in a first application of Montgomery's method, determining an intermediate result based on at least Montgomery's method for the modular operation, the first operand, and the first constant; and

in a second application of Montgomery's method, determining and storing in memory a final result that comprises the at least one part of the second electronic message, based on at least Montgomery's method for the modular operation, the intermediate result, and the second operand.

35. (Currently Amended) An apparatus for encryption and decryption of electronic messages based on an encryption protocol, comprising:

means for receiving a first electronic message that is encrypted according to the encryption protocol;

means for generating at least one part of a second electronic message, based on at least the first electronic message, a modular operation that is based on two applications of Montgomery's method, a first operand, a second operand, a modulus, and a modular reduction using a negative multiplicative inverse of the modulus, and wherein the means for generating the second electronic message further comprises:

means for generating a first constant (R) based on the modulus (M) by selecting a second constant (W) such that W ≥ 4M and determining the first constant (R) according to the expression $R = W^2$ (mod(M));[[;]]

means for determining, in a first application of Montgomery's method, an intermediate result based on at least Montgomery's method for the modular operation, the first operand, and the first constant; and

means for determining, in a second application of Montgomery's method, and storing in memory a final result that comprises the at least one part of the second electronic message, based on at least Montgomery's method for the modular operation, the intermediate result, and the second operand.

36. (Currently Amended) An apparatus for encryption and decryption of electronic messages based on an encryption protocol, comprising:

an interface;

a processor coupled to the interface and receiving information from the interface; and

one or more stored sequences of instructions which, when executed by the processor, cause the processor to carry out the steps of:

receiving a first electronic message that is encrypted according to the encryption protocol;

generating at least one part of a second electronic message, based on at least the first electronic message, a modular operation that is based on two applications of Montgomery's method, a first operand, a second operand, a modulus, and a modular reduction using a negative multiplicative inverse of the modulus, and wherein the instructions for generating the second electronic message further comprise instructions which, when executed by the processors, cause the processor to carry out the steps of:

generating a first constant (R) based on the modulus (M) by selecting a second constant (W) such that $W \geq 4M$ and determining the first constant (R) according to the expression $R = W^2$ (mod(M));[[;]]

in a first application of Montgomery's method, determining an intermediate result based on at least Montgomery's method for the modular operation, the first operand, and the first constant; and

in a second application of Montgomery's method, determining and storing in memory a final result that comprises the at least one part of the second electronic message, based on at least Montgomery's method for the modular operation, the intermediate result, and the second operand.

37. (Cancelled)

38. (Previously Presented) The apparatus of Claim 36, wherein the second constant (W) is not a power of two.

39. (Previously Presented) The apparatus of Claim 36, wherein the modular operation for determining the intermediate result is a modular multiplication that is based on at least the first operand, the first constant, a second constant, the modulus, and the negative multiplicative inverse of the modulus, and wherein the instructions for determining the intermediate result further comprise one or more stored sequences of instructions which, when executed by the processor, cause the processor to carry out the steps of:

determining a modified first operand based on the first operand and the first constant;

determining a modular reduction of the modified first operand based on the modified first operand, the negative multiplicative inverse of the modulus, and the second constant; and

determining the intermediate result based on the modified first operand, the modular reduction of the modified first operand, the modulus, and the second constant.

40. (Previously Presented) The apparatus of Claim 36, wherein the modular operation for determining the final result is a modular multiplication that is based on at least the second operand, a second constant, the modulus, the negative multiplicative inverse of the modulus, and the intermediate result, and wherein the instructions for determining the final result further comprise one or more stored sequences of instructions which, when executed by the processor, cause the processor to carry out the steps of:

determining a modified second operand based on the second operand and the intermediate result;

determining a modular reduction of the modified second operand based on the modified second operand, the negative multiplicative inverse of the modulus, and the second constant; and

determining the final result based on the modified second operand, the modular reduction of the modified second operand, the modulus, and the second constant.

41. (Previously Presented) The apparatus of Claim 36, wherein the modular operation for determining the intermediate result is a modular exponentiation that is based on at least the first operand, the first constant, a second constant, the modulus, and the negative multiplicative inverse of the modulus, and wherein the instructions for determining the intermediate result further comprise one or more stored sequences of instructions which, when executed by the processor, cause the processor to carry out the steps of:

determining a modified first operand based on the first operand and the intermediate result;

determining a modular reduction of the modified second operand based on the modified second operand, the negative multiplicative inverse of the modulus, and the second constant; and

determining the intermediate result based on the modified second operand, the modular reduction of the modified second operand, the modulus, and the second constant.

42. (Currently Amended) The apparatus of Claim 36, wherein the modular operation for determining the final result is a modular exponentiation that is based on at least the second operand, a second constant, the modulus, ~~a~~ the negative multiplicative inverse of the modulus, and the intermediate result, wherein the second operand includes a plurality of digits, and wherein the instructions for determining the final result further comprise one or more stored sequences of instructions which, when executed by the processor, cause the processor to carry out the steps of:

specifying a previous final result as having a value of one and a previous intermediate result as the intermediate result;

for each digit of the plurality of digits included in the second operand, carrying out the steps of:

when each digit of the plurality of digits has the value of one, then carrying out the steps of:

determining an updated final result based on the previous final result and the previous intermediate result;

determining a modular reduction of the updated final result based on the updated final result, the negative multiplicative inverse of the modulus, and the second constant;

determining a revised updated final result based on the updated final result, the modular reduction of the final result, the modulus, and the second constant;

if all digits of the plurality of digits have not been evaluated, specifying the revised updated final result as the previous final result; and

if all digits of the plurality of digits have been evaluated, specifying the revised updated final result as the final result;

determining an updated intermediate result based on the previous intermediate result;

determining a modular reduction of the updated intermediate result based on the updated intermediate result, the negative multiplicative inverse of the modulus, and the second constant;

determining a revised updated intermediate result based on the updated intermediate result, the modular reduction of the updated intermediate result, the modulus, and the first constant; and

specifying the revised updated intermediate result as the previous intermediate result.

43. (Previously Presented) The apparatus of Claim 36, further comprising one or more stored sequences of instructions which, when executed by the processor, cause the processor to carry out the steps of:

generating a plurality of residual number system (RNS) representations, wherein the plurality of RNS representations includes at least one RNS representation for each of the first operand, the modulus, and the first constant;

wherein the instructions for determining the intermediate result one or more stored sequences of instructions which, when executed by the processor, cause the processor to carry out the step of generating the intermediate result based on Montgomery's method for the modular operation and the plurality of RNS representations; and

wherein the instructions for determining the final result includes one or more stored sequences of instructions which, when executed by the processor, cause the processor to carry out the step of generating the final result based on Montgomery's method for the modular operation and the plurality of RNS representations.

44. (Previously Presented) The apparatus of Claim 43, wherein:

the plurality of RNS representations includes a first set of RNS representations in a first RNS base and a second set of RNS representations in a second RNS base;

the first RNS base extends the second RNS base;

the first RNS base includes a first group of sixty-four residues and the second RNS base includes a second group of sixty-four residues;

each residue in both the first group and second group is a seventeen-bit residue;

each residue in the first group is relatively prime with respect to all other residues in the first group and each residue in the second group is relatively prime with respect to all other residues in the second group;

each residue in both the first group and the second group are a selected from a range of 2^{16} to 2^{17}; and

further comprising one or more stored sequences of instructions which, when executed by the processor, cause the processor to carry out the steps of:

converting, in eight clock cycles, a first RNS representation of the first set of RNS representations in the first RNS base to a second RNS representation of the second set of RNS representations in the second RNS base;

performing operations involving each residue in the first group in parallel; and

performing operations involving each residue in the second group in parallel.

45. (Previously Presented) The apparatus of Claim 36, wherein the modular operation is a modular multiplication and the instructions for generating the second electronic message further comprises one or more stored sequences of instructions which, when executed by the processor, cause the processor to carry out the step of:

while determining the intermediate result and determining the final result, storing results of intermediate computations in a first register file and a second register file, wherein the first register file includes a first set of sixty-four seventeen-bit registers and the second register file includes a second set of sixty-four seventeen-bit registers.

46. (Previously Presented) The apparatus of Claim 36, wherein the modular operation is a modular exponentiation and the instructions for generating the second electronic message further comprises one or more stored sequences of instructions which, when executed by the processor, cause the processor to carry out the step of:

while determining the intermediate result and determining the final result, storing results of intermediate computations, in a first register file a second register file, a third register file, and a fourth register file; and

wherein the first register file includes a first set of sixty-four seventeen-bit registers, the second register file includes a second set of sixty-four seventeen-bit registers, the third register file includes a third set of sixty-four seventeen-bit registers, and the fourth register file includes a fourth set of sixty-four seventeen-bit registers.

47. (Previously Presented) The apparatus of Claim 36, wherein the instructions for determining the intermediate result and the instructions for determining the final result make use of an array of sixty-four seventeen-bit by seventeen-bit modular multiplier circuits, and wherein the array of sixty-four seventeen-bit by seventeen-bit modular multiplier circuits includes a plurality of ratio four to two compressors that are organized into three levels and that are executed in one clock cycle.

48. (Previously Presented) The apparatus of Claim 36, wherein the instructions for determining the intermediate result and the instructions for determining the final result make use of an array of sixty-four thirty-four-bit to seventeen-bit modular reduction circuits that are executed in one clock cycle.

49. (Cancelled)

50. (Previously Presented) The computer-readable medium of Claim 34, wherein the second constant (W) is not a power of two.

51. (Previously Presented) The computer-readable medium of Claim 34, wherein the modular operation for determining the intermediate result is a modular multiplication that is based on at least the first operand, the first constant, a second constant, the modulus, and the negative multiplicative inverse of the modulus, and wherein the instructions for determining the intermediate result further comprise one or more sequences of instructions, which when executed by one or more processors, cause the one or more processors to carry out the steps of:

determining a modified first operand based on the first operand and the first constant;

determining a modular reduction of the modified first operand based on the modified first operand, the negative multiplicative inverse of the modulus, and the second constant; and

determining the intermediate result based on the modified first operand, the modular reduction of the modified first operand, the modulus, and the second constant.

52. (Previously Presented) The computer-readable medium of Claim 34, wherein the modular operation for determining the final result is a modular multiplication that is based on at least the second operand, a second constant, the modulus, the negative multiplicative inverse of the modulus, and the intermediate result, and wherein the instructions for determining the final result further comprise one or more sequences of instructions, which when executed by one or more processors, cause the one or more processors to carry out the steps of:

determining a modified second operand based on the second operand and the intermediate result;

determining a modular reduction of the modified second operand based on the modified second operand, the negative multiplicative inverse of the modulus, and the second constant; and

determining the final result based on the modified second operand, the modular reduction of the modified second operand, the modulus, and the second constant.

53. (Previously Presented) The computer-readable medium of Claim 34, wherein the modular operation for determining the intermediate result is a modular exponentiation that is based on at least the first operand, the first constant, a second constant, the modulus, and the negative multiplicative inverse of the modulus, and wherein the instructions for determining the intermediate result further comprise one or more sequences of instructions, which when executed by one or more processors, cause the one or more processors to carry out the steps of:

determining a modified first operand based on the first operand and the intermediate result;

determining a modular reduction of the modified second operand based on the modified second operand, the negative multiplicative inverse of the modulus, and the second constant; and

determining the intermediate result based on the modified second operand, the modular reduction of the modified second operand, the modulus, and the second constant.

54. (Previously Presented) The computer-readable medium of Claim 34, wherein the modular operation for determining the final result is a modular exponentiation that is based on at least the second operand, a second constant, the modulus, the negative multiplicative inverse of the modulus, and the intermediate result, wherein the second operand includes a plurality of digits, and wherein the instructions for determining the final result further comprise one or more sequences of instructions, which when executed by one or more processors, cause the one or more processors to carry out the steps of:

specifying a previous final result as having a value of one and a previous intermediate result as the intermediate result;

for each digit of the plurality of digits included in the second operand, carrying out the steps of:

when each digit of the plurality of digits has the value of one, then carrying out the steps of:

determining an updated final result based on the previous final result and the previous intermediate result;

determining a modular reduction of the updated final result based on the updated final result, the negative multiplicative inverse of the modulus, and the second constant;

determining a revised updated final result based on the updated final result, the modular reduction of the final result, the modulus, and the second constant;

if all digits of the plurality of digits have not been evaluated, specifying the revised updated final result as the previous final result; and

if all digits of the plurality of digits have been evaluated, specifying the revised updated final result as the final result;

determining an updated intermediate result based on the previous intermediate result;

determining a modular reduction of the updated intermediate result based on the updated intermediate result, the negative multiplicative inverse of the modulus, and the second constant;

determining a revised updated intermediate result based on the updated intermediate result, the modular reduction of the updated intermediate result, the modulus, and the first constant; and

specifying the revised updated intermediate result as the previous intermediate result.

55. (Previously Presented) The computer-readable medium of Claim 34, further comprising one or more sequences of instructions, which when executed by one or more processors, cause the one or more processors to carry out the step of:

generating a plurality of residual number system (RNS) representations, wherein the plurality of RNS representations includes at least one RNS representation for each of the first operand, the modulus, and the first constant;

wherein the instructions for determining the intermediate result includes one or more sequences of instructions, which when executed by one or more processors, cause the one or more processors to carry out the step of generating the intermediate result based on Montgomery's method for the modular operation and the plurality of RNS representations; and

wherein the instructions for determining the final result includes one or more sequences of instructions, which when executed by one or more processors, cause the one or more processors to carry out the step of generating the final result based on Montgomery's method for the modular operation and the plurality of RNS representations.

56. (Previously Presented) The computer-readable medium of Claim 55, wherein:

the plurality of RNS representations includes a first set of RNS representations in a first RNS base and a second set of RNS representations in a second RNS base;

the first RNS base extends the second RNS base;

the first RNS base includes a first group of sixty-four residues and the second RNS base includes a second group of sixty-four residues;

each residue in both the first group and second group is a seventeen-bit residue;

each residue in the first group is relatively prime with respect to all other residues in the first group and each residue in the second group is relatively prime with respect to all other residues in the second group;

each residue in both the first group and the second group are a selected from a range of 2^{16} to 2^{17}; and

further comprising one or more sequences of instructions, which when executed by one or more processors, cause the one or more processors to carry out the steps of:

converting, in eight clock cycles, a first RNS representation of the first set of RNS representations in the first RNS base to a second RNS representation of the second set of RNS representations in the second RNS base;

performing operations involving each residue in the first group in parallel; and

performing operations involving each residue in the second group in parallel.

57. (Previously Presented) The computer-readable medium of Claim 34, wherein the modular operation is a modular multiplication and the instructions for generating the second electronic message further comprises one or more sequences of instructions, which when executed by one or more processors, cause the one or more processors to carry out the step of:

while determining the intermediate result and determining the final result, storing results of intermediate computations in a first register file and a second register file, wherein the first register file includes a first set of sixty-four seventeen-bit registers and the second register file includes a second set of sixty-four seventeen-bit registers.

58. (Previously Presented) The computer-readable medium of Claim 34, wherein the modular operation is a modular exponentiation and the instructions for generating the second electronic message further comprises one or more sequences of instructions, which when executed by one or more processors, cause the one or more processors to carry out the step of:

while determining the intermediate result and determining the final result, storing results of intermediate computations, in a first register file a second register file, a third register file, and a fourth register file; and

wherein the first register file includes a first set of sixty-four seventeen-bit registers, the second register file includes a second set of sixty-four seventeen-bit registers, the third register file includes a third set of sixty-four seventeen-bit registers, and the fourth register file includes a fourth set of sixty-four seventeen-bit registers.

59. (Previously Presented) The computer-readable medium of Claim 34, wherein the instructions for determining the intermediate result and the instructions for determining the final result make use of an array of sixty-four seventeen-bit by seventeen-bit modular multiplier circuits, and wherein the array of sixty-four seventeen-bit by seventeen-bit modular multiplier circuits includes a plurality of ratio four to two compressors that are organized into three levels and that are executed in one clock cycle.

60. (Previously Presented) The computer-readable medium of Claim 34, wherein the instructions for determining the intermediate result and the instructions for determining the final result make use of an array of sixty-four thirty-four-bit to seventeen-bit modular reduction circuits that are executed in one clock cycle.

61. (Cancelled)

62. (Previously Presented) The apparatus of Claim 35, wherein the second constant (W) is not a power of two.

63. (Previously Presented) The apparatus of Claim 35, wherein the modular operation for determining the intermediate result is a modular multiplication that is based on at least the first operand, the first constant, a second constant, the modulus, and ~~a~~ the negative multiplicative inverse of the modulus, and wherein the means for determining the intermediate result further comprises:

means for determining a modified first operand based on the first operand and the first constant;

means for determining a modular reduction of the modified first operand based on the modified first operand, the negative multiplicative inverse of the modulus, and the second constant; and means for determining the intermediate result based on the modified first operand, the modular reduction of the modified first operand, the modulus, and the second constant.

64. (Previously Presented) The apparatus of Claim 35, wherein the modular operation for determining the final result is a modular multiplication that is based on at least the second operand, a second constant, the modulus, the negative multiplicative inverse of the modulus, and the intermediate result, and wherein the means for determining the final result further comprises:

means for determining a modified second operand based on the second operand and the intermediate result;

means for determining a modular reduction of the modified second operand based on the modified second operand, the negative multiplicative inverse of the modulus, and the second constant; and

means for determining the final result based on the modified second operand, the modular reduction of the modified second operand, the modulus, and the second constant.

65. (Previously Presented) The apparatus of Claim 35, wherein the modular operation for determining the intermediate result is a modular exponentiation that is based on at least the first operand, the first constant, a second constant, the modulus, and the negative multiplicative inverse of the modulus, and wherein the means for determining the intermediate result further comprises:

means for determining a modified first operand based on the first operand and the intermediate result;

means for determining a modular reduction of the modified second operand based on the modified second operand, the negative multiplicative inverse of the modulus, and the second constant; and

means for determining the intermediate result based on the modified second operand, the modular reduction of the modified second operand, the modulus, and the second constant.

66. (Previously Presented) The apparatus of Claim 35, wherein the modular operation for determining the final result is a modular exponentiation that is based on at least the second operand, a second constant, the modulus, the negative multiplicative inverse of the modulus, and the intermediate result, wherein the second operand includes a plurality of digits, and wherein the means for determining the final result further comprises:

means for specifying a previous final result as having a value of one and a previous intermediate result as the intermediate result;

means for performing, for each digit of the plurality of digits included in the second operand, the following steps:

when each digit of the plurality of digits has the value of one, then performing the computer-implemented steps of:

determining an updated final result based on the previous final result and the previous intermediate result;

determining a modular reduction of the updated final result based on the updated final result, the negative multiplicative inverse of the modulus, and the second constant;

determining a revised updated final result based on the updated final result, the modular reduction of the final result, the modulus, and the second constant;

if all digits of the plurality of digits have not been evaluated, specifying the revised updated final result as the previous final result; and

if all digits of the plurality of digits have been evaluated, specifying the revised updated final result as the final result;

determining an updated intermediate result based on the previous intermediate result;

determining a modular reduction of the updated intermediate result based on the updated intermediate result, the negative multiplicative inverse of the modulus, and the second constant;

determining a revised updated intermediate result based on the updated intermediate result, the modular reduction of the updated intermediate result, the modulus, and the first constant; and

specifying the revised updated intermediate result as the previous intermediate result.

67. (Previously Presented) The apparatus of Claim 35, further comprising:

means for generating a plurality of residual number system (RNS) representations, wherein the plurality of RNS representations includes at least one RNS representation for each of the first operand, the modulus, and the first constant;

wherein the means for determining the intermediate result includes means for generating the intermediate result based on Montgomery's method for the modular operation and the plurality of RNS representations; and

wherein the means for determining the final result includes means for generating the final result based on Montgomery's method for the modular operation and the plurality of RNS representations.

68. (Previously Presented) The apparatus of Claim 67, wherein:

the plurality of RNS representations includes a first set of RNS representations in a first RNS base and a second set of RNS representations in a second RNS base;

the first RNS base extends the second RNS base;

the first RNS base includes a first group of sixty-four residues and the second RNS base includes a second group of sixty-four residues;

each residue in both the first group and second group is a seventeen-bit residue;

each residue in the first group is relatively prime with respect to all other residues in the first group and each residue in the second group is relatively prime with respect to all other residues in the second group;

each residue in both the first group and the second group are a selected from a range of 2^{16} to 2^{17}; and further comprising:

means for converting, in eight clock cycles, a first RNS representation of the first set of RNS representations in the first RNS base to a second RNS representation of the second set of RNS representations in the second RNS base;

means for performing operations involving each residue in the first group in parallel; and

means for performing operations involving each residue in the second group in parallel.

69. (Previously Presented) The apparatus of Claim 35, wherein the modular operation is a modular multiplication and the means for generating the second electronic message further comprises:

means for storing, while determining the intermediate result and determining the final result, results of intermediate computations in a first register file and a second register file, wherein the first register file includes a first set of sixty-four seventeen-bit registers and the second register file includes a second set of sixty-four seventeen-bit registers.

70. (Previously Presented) The apparatus of Claim 35, wherein the modular operation is a modular exponentiation and the means for generating the second electronic message further comprises:

means for storing, while determining the intermediate result and determining the final result, results of intermediate computations, in a first register file a second register file, a third register file, and a fourth register file; and

wherein the first register file includes a first set of sixty-four seventeen-bit registers, the second register file includes a second set of sixty-four seventeen-bit registers, the third register file includes a third set of sixty-four seventeen-bit registers, and the fourth register file includes a fourth set of sixty-four seventeen-bit registers.

71. (Previously Presented) The apparatus of Claim 35, wherein the means for determining the intermediate result and the means for determining the final result use an array of sixty-four seventeen-bit by seventeen-bit modular multiplier circuits, and wherein the array of sixty-four seventeen-bit by seventeen-bit modular multiplier circuits includes a plurality of ratio four to two compressors that are organized into three levels and that are executed in one clock cycle.

72. (Previously Presented) The apparatus of Claim 35, wherein the means for determining the intermediate result and the means for determining the final result use an array of sixty-four thirty-four-bit to seventeen-bit modular reduction circuits that are executed in one clock cycle.